УДК 004.82+007.52

О.О. Варламов, Р.А. Санду, А.Н. Владимиров, А.Ю. Бадалов, К.Э. Тожа
ООО «МИВАР», Московский автомобильно-дорожный государственный технический университет (МАДИ), ФГУП Научно-исследовательский институт радио, г. Москва
info@mivar.ru, ovar@mivar.ru, ovar@narod.ru

Миварный метод логико-вычислительной обработки информации для АСУ, тренажеров и экспертных систем реального времени

В статье разработан новый быстрый метод унифицированной логико-вычислительной обработки информации, основанный на синтезе и развитии продукционного подхода и миварных сетей. На основе миварного описания предметной области этот метод может использоваться для создания различных автоматизированных или автоматических систем управления, экспертных систем, включая и обработку в реальном времени, тренажеров, архитектур, основанных на сервисах, и других автоматизированных систем обработки информации.

Введение

Для решения многих практических задач применения информационных систем и процессов требуется проводить как логическую, так и вычислительную обработку данных. Исторически так сложилось, что области логического вывода и вычислительной обработки развивались самостоятельно и успешно решали различные классы задач. В некотором смысле даже существовало противоречие между этими разными подходами. В настоящее время возникли более сложные и комплексные задачи. С другой стороны, и возможности современных компьютеров значительно увеличились. Комплексные сложные задачи существовали всегда, но не было адекватных методов их решения. Если проводить аналогию с человеком, то наш разум одновременно комплексно решает и логические, и вычислительные, и логико-вычислительные задачи. Что же мешает нам в XXI веке автоматизировать эти процессы? Данная работа обосновывает, что новые методы обработки информации позволяют снять многие ограничения и выполнять комплексную логико-вычислительную обработку данных на основе продукций и миварной сети правил. Для этого надо разработать миварный метод логико-вычислительной обработки для решения достаточно большого класса сложных научных и практических задач.

Целью данной работы является разработка миварного метода логико-вычислительной обработки информации для различных АСУ, тренажеров, экспертных систем реального времени, архитектур, ориентированных на сервисы, а также для других автоматизированных систем обработки информации.

Прежде всего, необходимо провести анализ существовавших ранее подходов к решению различных классов задач и оценить их ограничения.

Анализ подходов к логической обработке и выводу

Под логической обработкой принято понимать некий вывод, лежащий в основе человеческих рассуждений. У Д.А. Поспелова отмечено: «Всякий вывод, как бы он не был организован, носит переборный характер. ...Повышение эффективности процесса вывода – центральная проблема всех автоматизированных систем дедуктивного вы-

вода» [1, с. 79]. Отметим, что многие ученые поддерживают это мнение [2], хотя существуют и другие взгляды на эту проблему [3-9]. Впрочем, все согласны с необходимостью повышения эффективности процесса вывода.

Известно описание общей схемы выводов, лежащей «в основе большого количества моделей человеческих достоверных рассуждений» [1, с. 83]. «Схема вывода не обязательно описывается в виде дерева. Она может иметь вид произвольной сети, ориентированной, неориентированной или частично ориентированной» [1, с. 84]. На рис. 1 показан пример неориентированной сети, аналогичный рисунку в [1, с. 83]. «Такая сеть ...называется И-ИЛИ сетью. Процесс вывода на И-ИЛИ сети протекает следующим образом. Пусть мы хотим доказать утверждение F6. В качестве априорно доказанного заданного утверждение F1. Как из F1 можно получить F6? Если считать, что все связи допускают ориентацию в нужную сторону, то из F1 можно получить F3, затем F5 и, наконец, F6. Но этот путь нам удалось отыскать потому, что сеть... мы видим “с высоты птичьего полета”. Лабиринт поиска лежит в виде чертежа перед нами. Именно это позволяет нам не делать лишних попыток, не двигаться в ненужную сторону, а идти кратчайшим путем к цели» [1, с. 84]. Отметим, что на рис. 1 изображен однодольный граф, т.е. все вершины графа принадлежат одному классу [2, с. 125]. Этот факт будет очень важен для дальнейшего анализа.

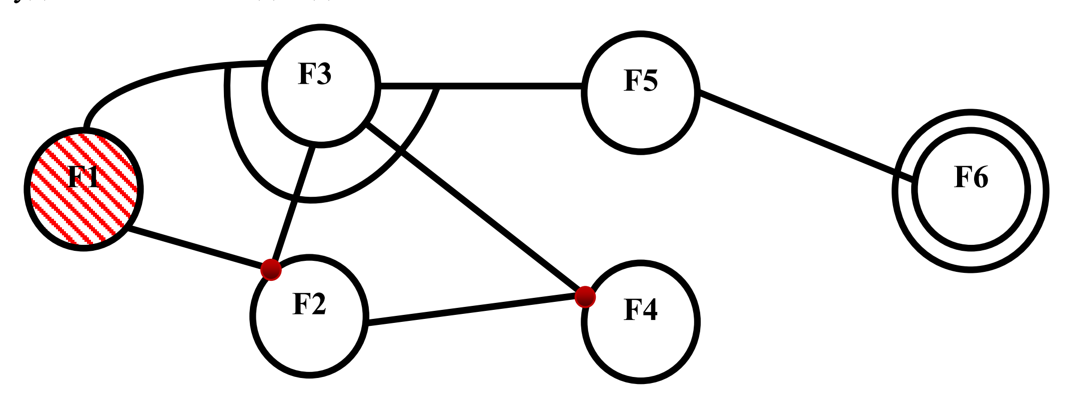


Рисунок 1 – Однодольный граф логического вывода

Д.А. Поспелов так описывает метод прямой волны логического вывода: «...мы описали процедуру, которая... носит универсальный характер и пригодна для поиска пути вывода в лабиринтах произвольного типа. Эта процедура известна... под названием метода прямой волны. Волна поиска путей к целевой площадке распространяется от всех площадок, играющих роль начальных» [1, с. 85]. Далее: «Возможен и другой способ поиска доказательства. Он носит название метода обратной волны. В этом методе волна начинает свое движение от целевых площадок и движется в направлении начальных площадок лабиринта. ...Часто используется смешанный метод вывода, при котором одновременно движутся прямая и обратная волны. При встрече этих волн формируется путь вывода от начальных аксиом к целевым выражениям» [1, с. 85]. Отметим, что эти работы не нашли широкого практического применения в прошлом веке. Это обусловлено тем, что не было предложено метода быстрой логической обработки данных. Как было показано выше, ученые исходили из предпосылки, что любой вывод носит переборный характер, т.е. имеет место факториальный рост вычислительной сложности при увеличении количества переменных. Даже современные суперкомпьютеры при таком подходе позволяют обрабатывать в близком к реальному времени масштабе лишь 10 – 15 переменных. Фактически это позволяет решать только «игрушечные» модельные задачи, т.к. современные требования начинаются с сотен и достигают десятков тысяч переменных даже для относительно простых предметных областей.

Для миварного метода логико-вычислительной обработки данных важно следующее: «...ядром всех основных типов рассмотренных интеллектуальных систем являются база знаний и блок, осуществляющий вывод с помощью знаний (решатель, планировщик или логический блок). Этот вывод составляет основную процедуру, реализуемую в интеллектуальных системах» [1, с. 129].

Возможности и ограничения продукционного подхода

В настоящее время продолжается дискуссия о роли и возможных применениях различных логических механизмов и, прежде всего, исчислении высказываний, исчислении предикатов и продукциях. У продукционного подхода есть важные преимущества. «Знания о внешнем мире могут иметь двоякую природу. Они могут содержать декларативное описание фактов и явлений внешнего мира, фиксирующее их наличие или отсутствие, а также основные связи и закономерности, в которые эти факты и явления входят. Но они могут содержать и процедурные описания того, как надо манипулировать с этими фактами и достигать целей, интересных для системы. Для описания знаний в интеллектуальных системах используются специальные языки описания знаний (ЯОЗ). ...Простейшими видами таких ЯОЗ являются языки исчисления высказываний или исчисления предикатов вместе с теми процедурами вывода, которые для них известны. Однако в современных интеллектуальных системах такие языки используются довольно редко. Куда более распространены в них языки, основанные на продукциях. Продукции в общем виде можно записать в форме “Если..., то...”. Сама по себе эта форма оказывается весьма характерной для фиксации знаний в различных областях человеческой деятельности. ...представление фрагментов наших знаний о внешнем мире и действиях в нем в виде продукций имеет весьма большое распространение. Часть специалистов по интеллектуальным системам считает, что запись знаний в виде систем продукций носит универсальный характер – любые знания можно записать в такой форме» [1, с. 129]. Отметим, что О.П. Кузнецов в [2, с. 282-283] под продукциями понимает множество правил вывода в канонических системах (системах продукций Поста), в которых есть посылки и следствия.

С точки зрения анализа метода логико-вычислительной обработки данных, принципиально важным является то, что в системе продукций можно представлять самые разнообразные правила, процедуры, формулы или сервисы. По этому поводу у Д.А. Поспелова написано: «приводят немало примеров, когда знания, внешне не имеющие продукционной формы, удается перевести в систему продукций» [1, с. 129-130]. Далее: «продукциями являются не только те выражения, которые имеют форму “Если..., то..”, но и многие другие выражения. К ним, по сути, сводятся все каузальные, т.е. причинно-следственные утверждения...» [1, с. 130]. Д.А. Поспелов делает вывод: «Продукционные системы получили при представлении знаний в последнее время наибольшее распространение» [1, с. 129]. Там же показаны 9 типов продукций и специально подчеркнуто, что возможны и другие [1, с. 131-134]. Далее делается вывод: «...продукции могут иметь весьма различное значение. В качестве их левых и правых частей могут выступать и некоторые утверждения, и действия» [1, с. 134]. Следовательно, применение продукционного подхода для логико-вычислительной обработки разнообразных данных является обоснованным и целесообразным. Отметим, что существует аналогичный подход, основанный на гиперправилах с мультиактивизаторами [3]. В.Ф. Хорошевский в [10, с. 82-83] при описании «слоенного пирога» Semantic Web выделяет промежуточный «слой правил», для которого ведутся исследования различных систем вывода на правилах.

Представление сетей продукций двудольными графами

Перейдем к формализованному представлению продукций и сетей, которые могут быть сформированы из них. Выше было отмечено, что Д.А. Поспелов [1, с. 83-84] представляет сеть правил в виде однодольного графа (рис. 1). Ранее не все ученые замечали аналогичность представления сетей правил и теории графов. В работах [3-6] предложен подход по взаимосвязи сетей правил и графов. Более того, показано, что некоторые задачи логического вывода можно решать на основе подходов теории графов. На данном этапе важно, что сети правил целесообразнее представлять в виде двудольных графов, получая нечто аналогичное сетям Петри, но с соответствующим развитием до миварных логических сетей [3-9].

Напомним определения двудольных графов. «Граф G = (V, E) называется двудольным, если существует разбиение V = {V1, V2} такое, что никакие две вершины из V1 или из V2 не являются смежными» [11, с. 223]. «Двудольным графом G = (X, Y, E) называется неориентированный граф, вершины которого можно разбить на два класса X и Y так, что концы каждого ребра принадлежат разным классам» [2, с. 125]. Приведем следующее важное замечание: «введенные понятия допускают естественное обобщение. Неориентированный граф называется k-дольным, если его вершины можно разбить на k классов так, что концы каждого ребра принадлежат разным классам» [2, с. 125].

Миварные логико-вычислительные сети

В России активно развивается новый подход к единому представлению знаний и данных, который получил название «миварный». Как показано в [11], миварный подход является обобщением и развитием продукционного подхода, сетей Петри и других формализмов, применяемых для логической обработки данных. Миварные сети могут быть представлены в виде двудольного графа, состоящего из «объектов-переменных» и «правил-процедур» (рис. 2). На рис. 2 «объекты» изображены овалами, «правила» изображены прямоугольниками, тонкими стрелками показаны связи «объектов» и «правил», а крупными стрелками показаны общие направления использования «объектов и «правил» для поиска логического вывода. Прежде всего, составляются два списка, которые и образуют «две непересекающиеся доли графа»: «объекты» и «правила». Каждое правило в миварной сети является развитием продукций, гиперправил с мультиактивизаторами или вычислительных процедур. Доказано в [3], что с точки зрения дальнейшей их обработки, все эти формализмы идентичны.

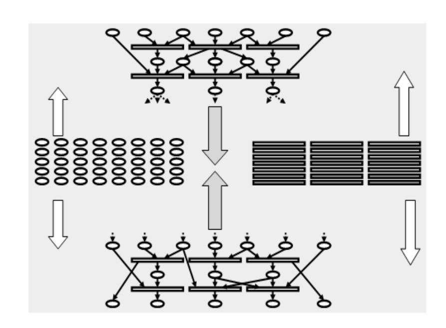


Рисунок 2 – Двудольный граф миварной логической сети

Теоретические основы миварного метода логико-вычислительной обработки информации («миварного метода») первоначально были разработаны в [3] и [4] еще в 2002 году. В статье [4] представлены теоретические основы создания линейного матричного метода определения маршрута логического вывода на адаптивной сети

правил. Именно этот метод и получил теперь новое название, более короткое и отражающее существо решаемой проблемы. Суть метода в том, что для сети правил, представленной в списочной форме, строится матрица. Затем путем анализа этой матрицы определяется факт наличия маршрута вывода, определяются возможные маршруты логического вывода и из этих маршрутов выбирают по заданным критериям оптимальности наиболее лучший, «кратчайший» маршрут. В статье [7] подробно и с «картинками» показаны все особенности работы этого метода для решения одной важной конкретной задачи в формализмах матриц и теории графов.

В работах [5], [6] и на сайтах [8], [9] изложены результаты практической реализации миварных сетей на основе создания программного комплекса «УДАВ». Этот комплекс реализует линейной вычислительной сложности миварный метод. Миварные сети позволяют реализовать активный обучаемый логический вывод для любых предметных областей, моделирование которых проводится в миварном информационном пространстве [3]. В настоящее время реализован универсальный метод «формирования алгоритмов» и его настройка для решения в реальном времени и при любых наборах входных данных «геометрических задач решений треугольников» с 33 объектами и 161 правилом (33!) [5], [6]. А на сайтах [8], [9] можно скачать демо-версию этого комплекса. Таким образом, в работах [3-9] доказано, что существует достаточно универсальный миварный подход и миварный метод для решения многих практических задач применения информационных систем и процессов с одновременным проведением как логической, так и вычислительной обработки данных. Следовательно, существовавшее ранее противоречие между логическим выводом и вычислительной обработкой успешно преодолено с помощью миварных логико-вычислительных сетей.

Описание миварного метода обработки информации

Для миварной сети логических правил, представленной в списочной форме, строится матрица. Затем, на основе анализа этой матрицы определяется факт наличия успешного маршрута вывода, потом определяются возможные маршруты логического вывода, а на последнем этапе из этих маршрутов выбирают «кратчайший», наиболее оптимальный по заданным критериям оптимальности. Итак, пусть известны m правил и n переменных (входящих в правила либо в качестве исходных, активизирующих, либо в качестве получаемых, т.е. выходных переменных). Тогда в матрице $V (n \times m)$ могут быть представлены все взаимосвязи между правилами и переменными. При этом, на каждой строке при описании каждого правила все входные переменные этого правила на соответствующих позициях матрицы помечаются символом x, все выходные – y, все переменные, которые уже получили в процессе вывода или задания исходных данных некоторое конкретное значение, – z, а все искомые (выходные) переменные, т.е. те, которые необходимо вывести из исходных (входных) данных, – w. Кроме того, добавим в матрицу V одну строку и один столбец для хранения в них служебной информации. Получаем матрицу V размерности $(n + 1) \times (m + 1)$, в которой отражена вся структура исходной сети правил. Сразу отметим, что структура этой логической сети может изменяться в любое время, т.е. это сеть правил с изменяемой (эволюционной) структурой. Опишем пример работы метода. Для поиска маршрута логического вывода на полученной матрице производят действия по следующим (пронумерованным) этапам.

1. В строке $(m + 1)$ помечают известные z и искомые w переменные.

2. Осуществляют последовательно, например, сверху вниз, поиск таких правил, которые могут быть активизированы, т.е. у которых известны все входные переменные. Если таких правил нет, то маршрута логического вывода нет и необходимо выдать

запрос на уточнение (добавление) входных данных. Если такие правила, которые могут быть активизированы, есть, то у каждого из них в соответствующем месте служебной строки делается пометка, что правило может быть запущено.

3. Если таких правил несколько, то осуществляется выбор по заранее определенным критериям такого или таких правил, которые должны быть активизированы в первую очередь. При наличии достаточных ресурсов, одновременно могут запускаться сразу несколько правил.

4. Имитация запуска правила (процедуры) осуществляется путем присваивания выводимым в этом правиле переменным значений «известно», т.е. в нашем примере – z. Запущенное правило, для удобства дальнейшей работы, помечается дополнительно, например, цифрой 2 (это не обязательно).

5. После имитации запуска правил проводят анализ достижения цели, т.е. анализируют получение требуемых значений путем сравнения служебных символов в служебной строке. Если в служебной строке (m + 1) осталось хоть одно значение «искомая» (т.е. w), то осуществляют дальнейший поиск маршрута логического вывода. В противном случае задача считается успешно решенной, а все задействованные правила в соответствующем порядке их запуска и образуют искомый маршрут логического вывода.

6. Прежде всего, определяют наличие таких правил, которые могут быть запущены после определения новых значений на предыдущем этапе. Если таких правил нет, то маршрута вывода нет и поступают аналогично пункту 2 этого метода. Если такие правила есть, то продолжают поиск маршрута вывода.

7. На следующем этапе опять аналогично этапу 4 запускают правила (имитируют запуск), далее аналогично этапам 5 и 6 выполняют необходимые действия столько раз, сколько требуется для получения результата. При необходимости повторяют все этапы (со 2 по 7) до достижения результата. При этом результат может быть как положительный – маршрут вывода существует, так и отрицательный – вывода нет из-за неопределенности входных данных. Для наглядности продолжим пошаговое выполнение нашего примера. Итак, в нашем примере необходимо провести имитацию запуска правила 2.

8. В клетках (4, m + 1) и (5, m + 1) получаем признак выводимости переменных 4 и 5, а в клетке (n + 1, 2) формируем признак того, что правило уже было запущено, т.е. ставим цифру 2. После этого проводим анализ служебной строки и видим, что не все искомые переменные известны. Значит необходимо продолжить обработку матрицы V размерности $(n + 1) \times (m + 1)$. Анализ этой матрицы показывает возможность запуска правила m.

9. Продолжим дальнейшую обработку примера. При запуске правила m появляются новые значения, в том числе и для искомых переменных.

10. Итак, в нашем примере в служебной строке больше не осталось искомых правил, а в клетках таблицы появились новые значения: в клетке (n + 1, m) – 2, а в клетке (n – 2, m + 1) вместо значения w появилось значение z. Таким образом, получен положительный результат, следовательно, маршрут логического вывода при данных исходных значениях существует.

Подчеркнем, что с научно-практической точки зрения основную сложность при использовании экспертных систем составляет именно концептуальное «продукционное» описание предметной области и формирование необходимых двух списков «объектов» и «правил» для миварных логических сетей. Непосредственная обработка проводится на основе универсального механизма, описанного в [3-9]. Выделенные правила-процедуры при необходимости могут быть представлены в виде продукций, что соответствует традиционному подходу [1], [2], [10], [11], но в то же время позволяет реализовать многомерное эволюционное представление данных. В то же время такое представление позволяет создать информационную модель нашей предметной области в форма-

лизме миварных сетей. В свою очередь, применение миварных сетей позволяет реализовать все специфические особенности решаемой в конкретный момент задачи. Таким образом, разработанный на основе продукций миварный метод логико-вычислительной обработки обеспечивает достижение сформулированной цели и осуществляет сложную и многоуровневую вычислительную обработку данных вместе с реализацией сложной логики расчетов и принятия решений.

Особенно подчеркнем, что вычислительная сложность метода пропорциональна произведению количества правил-процедур на количество объектов-переменных, т.е. является линейной по отношению к переменным. Это позволяет обрабатывать сложные практические задачи, например, по управлению инновационными ресурсами, в которых одновременно используется до десяти тысяч переменных, в реальном масштабе времени. При необходимости возможно эволюционное наращивание количества переменных и правил. Проведенные нами вычислительные эксперименты подтверждают выводы, полученные в работах [3-9].

Отметим, что данный метод может применяться в различных отраслях промышленности или для решения задач в масштабах всей экономики России и мира. Универсальность данного метода обусловлена применением продукций и миваров. Как было указано выше, «запись знаний в виде систем продукций носит универсальный характер – любые знания можно записать в такой форме» [1, с. 129]. Универсальные возможности миварного подхода обусловлены тем, что он обобщает все известные модели данных, включая семантические сети, «сущность-связь» и онтологии [3-9]. В свою очередь, миварные сети являются развитием продукционного подхода с переходом от однодольных графов к двудольным и многодольным и многомерным эволюционным графам. Следовательно, в формализме миварной сети правил можно описать любые данные, а также зависимости, функции и отношения в виде миварного многомерного эволюционного пространства унифицированного представления данных и правил.

Выводы

Для экспертных систем разработан быстрый Миварный метод логико-вычислительной обработки данных на основе продукций и миварной логической сети правил. Данный метод позволяет в едином формализме проводить и логическую обработку – вывод, и выполнять различные вычислительные процедуры с возможностью выработки управляющих воздействий или рекомендаций управленцам (экспертам). Вычислительная сложность разработанного метода является линейной по отношению к переменным, либо по отношению к правилам. Это позволяет обрабатывать сложные практические задачи с десятками тысяч переменных в реальном масштабе времени. Получено практическое подтверждение результатов работы метода в реальных условиях эксплуатации. Таким образом, разработан новый быстрый Миварный метод логико-вычислительной обработки информации для АСУ, тренажеров, экспертных систем реального времени и архитектур, ориентированных на сервисы, а также для других автоматизированных систем обработки информации.

Литература

1. Поспелов Д.А. Моделирование рассуждений. Опыт анализа мыслительных актов / Поспелов Д.А. – М. : Радио и связь, 1989. – 184 с.
2. Кузнецов О.П. Дискретная математика для инженера / Кузнецов О.П. – [6-е изд., стер.]. – СПб. : Лань, 2009. – 400 с.
3. Варламов О.О. Эволюционные базы данных и знаний для адаптивного синтеза интеллектуальных систем. Миварное информационное пространство / Варламов О.О.. – М. : Радио и связь, 2002. – 288 с.

4. Варламов О.О. Разработка линейного матричного метода определения маршрута логического вывода на адаптивной сети правил / О.О. Варламов // Известия вузов. Электроника. – 2002. – № 6. – С. 43-51.
5. Программный комплекс «УДАВ»: практическая реализация активного обучаемого логического вывода с линейной вычислительной сложностью на основе миварной сети правил / А.Н. Владимиров, О.О. Варламов, А.В. Носов, Т.С. Потапова // Труды НИИР : сб. ст. – М. : НИИР, 2010. – № 1. – С. 108-116.
6. About realization of linear computing complexity of the matrix method of route searching of the logic conclusion in the program complex «UDAV» on basis of mivarnoi network of rules / A.V. Nosov, A.N. Vladimirov, O.O. Varlamov [et al.] // Proceedings of the Congress on intelligent systems and information technologies «AIS-IT`09». Scientific publication in the 4 volumes. – Moscow : Physmatlit, 2009. – Vol. 4. – P. 63.
7. Санду Р.А. Метод логико-вычислительной обработки данных химической и нефтехимической промышленности России на основе продукций и миварной сети правил для управления инновационными ресурсами / Р.А. Санду // Прикладная информатика. – 2010. – № 3(27). – С. 87-107.
8. Веб-сайт д.т.н. Варламов О.О. [Электронный ресурс]. – Режим доступа : http://www.ovar.narod.ru. 2010.
9. Веб-сайт МИВАР [Электронный ресурс]. – Режим доступа : http://www.mivar.ru. 2010.
10. Хорошевский В.Ф. Пространства знаний в сети Интернет и Semantic Web (Часть 1) / В.Ф. Хорошевский // Искусственный интеллект. – 2008. – № 1. – С. 80-97.
11. Кук Д. Компьютерная математика / Д. Кук, Г. Бейз ; [пер. с англ.]. – М. : Наука, 1990. – 384 с.

О.О. Варламов, Р.А. Санду, А.М. Владимиров, А.Ю. Бадалов, К.Е. Тожа

Міварний метод логіко-обчислювальної обробки інформації для АСУ, тренажерів, експертних систем у режимі реального часу

У статті розглядається новий швидкий метод уніфікованої логіко-обчислювальної обробки інформації і даних на основі узагальнення і вироблення підходу виробництва і mivar-логічних мереж. Крім mivar або продукту опису предметної області, цей метод може бути використаний для створення різних автоматизованих і автоматичних систем управління, експертних систем, у тому числі обробки в режимі реального часу, тренажерів, архітектур, заснованих на сервісах, та інших автоматизованих систем обробки інформації.

O.O. Varlamov, R.A. Sandu, A.N. Vladimirov, A.Y. Badalov, K.E. Tozha

Mivar Method of Logical-computational Processing of Information for the ACS Simulators Real-time and Expert Systems

The article discusses a new rapid method for unified logical-computational processing of information and data, based on the synthesis and development of Production approach and mivar logical networks. After mivar or product descriptions subject area, this method can be used to create various automated or automatic control systems, expert systems, including the processing of real-time simulators, architectures based on services and other automated data processing systems.

Статья поступила в редакцию 19.07.2010.